

A NEW MATHEMATICAL RISK MANAGEMENT MODEL FOR AGILE SOFTWARE DEVELOPMENT METHODOLOGIES

Osaki Miller Thom-Manuel, Chidiebere Ugwu and Laeticia Nneka Onyejegbu

Department of Computer Science, University of Port Harcourt,
Port Harcourt, 500102, Nigeria

ABSTRACT

This paper proposes a new mathematical model for estimating the cost of explicit Agile software development risk management with its Impact Benefit s (savings/profits). This is necessitated by the fact that despite the increase in the need for managing risks explicitly in medium-to-large scale agile software development projects presently, there are no known ways to estimate explicit risk management costs/benefits. With the proposed model, explicit risk management procedures alongside with risk management estimation techniques is made known to Stakeholders who will be able to make the right decisions on risk management costs and its impacts as well as when to utilise implicit or explicit risk management. The proposed system proves to be feasible and dependable and is evidently capable of enhancing the agile methods for use for all sizes of software projects while still maintaining the swiftness of the agile process.

KEYWORDS

Requested Risk Management Budget, Assigned Risk Management Budget, Actual Risk Management Budget, UncontrolledLoss, Needed Cost of Control

1. INTRODUCTION

Software risk is the likelihood of suffering loss in software development. Software risk management is an important tool to checkmate such events/activities as wrong budget estimation, budget overrun, and project scope creep, continuous change in requirements, wrong schedule estimation and sometimes even outright project failure. It involves a set of practice and methods that follow a systematic order of identifying and analyzing risk and then taking precautionary measures to curb its occurrence or impact [1]. Furthermore, it is continuous and a must-follow practice in every software development process and project that must succeed. Consequently, it is considered a standard practice in the software industry [2] ; [3] ; [4] ; [5] .

Agile software development methodologies refer to software methodologies that conform to the philosophy, features and practices as stated in the agile manifesto. Agile methods by their nature depend on the communication between self-organized development teams and customer representatives to produce small releases of quality software at short time frames called iterations with user requirements constantly being capable of modification at any time in the development

process. This set of value attributes of agile methods make them capable of eliminating threats that very often generate to risky conditions such as the production of software with wrong features and hence poor quality software, budget overruns and wrong schedule estimation. This, however, is achievable in small software projects where the number of iterations is small but for medium-to-large and/or complex software development projects where the number of iterations is many, conscious and formal risk management tools and techniques become necessary. Evidences to this fact are seen in the statements of some agile proponents and other researchers namely [6], [7] and [8] that agile methods are suitable for simple small or low-risk software projects and some form of explicit risk management is needed if they are to be used for the development of large software projects. Furthermore, in the study of [9] it was stated that Agile practices by themselves are not sufficient to address the risks that impact most medium-to-large software projects. While this and many similar study results indicate that explicit risk management is necessary if agile software methodologies is to be used for medium-to-large scale software projects, there is also the fear that the cost of implementing explicit risk management will be expensive as risk management is a project on its own and as such would incur additional costs. Though, [6] stated that explicitly managing risks is costlier than implicitly managing risk and another statement by [10] indicates that the cost of dealing with an actual risk is normally far more than the cost of implementing risk management. These statements are without proof as there is no known way to estimate the cost of explicit risk management processes.

Notwithstanding, the Risk cost budget estimation procedure which is a sub-cost of an entire risk management process budget has been implemented by [11] in a Mathematical Risk Management Model (MRMM) developed for managing risk in iterative software development projects specifically agile software projects. MRMM is built specially to take care of people behaviour related risks. It outlines nine risk management plans to be followed in agile software projects to achieve a quick and organised risk management process. The steps includes risk identification, risk analysis, risk cost evaluation, Total risk amount calculation, total risk amount calibration, Requested risk budget, Real risk budget, Risk removal process and Iteration result analysis. It has Computer Aided Software Engineering (CASE) tools namely Agile Risk Information Sheet (ARIS) for taking down risk data during agile team meetings and a relational database for storing data recorded in ARIS.

2. REVIEW OF RELATED WORKS

Although there exist few risk management models developed for use in the agile methods, each of these models is built to suit a particular agile model and with or without some level/aspect of risk management estimation tool as in [8] that proposed the implementation of risk management with Scrum to achieve the requirement of Capability Maturity Model I (CMMI) and thus improve software quality via the improvement of the software development process. This they actualized by integrating a risk register into the SCRUM model. Though helpful in dealing with risk before the deployment stage of each sprint and in sprint retrospective meetings, it is not ideal for use in large projects and does not provide any estimation tool for risk management.

[12] proposed an integration process model between the PRINCE2 and SCRUM by embedding the project risk management procedure into the SCRUM framework to prove and resolve the lack of risk management procedure in the agile development. This also does not address other important issues like risk management estimation.

Though, a risk management tool developed by [13] formulated ways of calculating risk control cost and its benefits, it was built specifically for extreme programming and risk management budget as determined does not include all budgets involved in explicit risk management. Other studies by [2] resulted in the creation of an integration model and integrated model which were intended to provide basic guidelines for integrating the risk management and agile processes and serve as a reference model against which software organizations can compare their risk management processes respectively. Here again, risk management and its benefits' estimation on the Agile development process were not considered.

The study by [10] resulted in the development of a risk management model that is mathematical for agile models. Apart from being detailed, it incorporated a risk cost estimation tool. This is suitable for estimating what it will take to rework task(s) affected by a particular risk if the risk eventually occurs. Nevertheless, risk cost is only a sub-cost of the entire risk management cost.

However, for any risk management process to be implemented explicitly, insights on how to estimate its cost as a whole should be known. In an earlier study by the present researchers, [1], an extended agile software development project budget model that incorporated explicit risk management expenses into the agile software development project budget was proposed. This, the researchers achieved by first spelling out the various activities that will be performed during an explicit risk management process and the likely expenses associated with them. These then formed the constituents of the explicit risk management expenses in the agile software development project. A model was developed in that regard to help stakeholders on the path to follow to successfully estimate risk management explicitly in all agile projects. This is a first step to actualizing a formal risk management process with the agile methods. This paper is proposing a new mathematical model for estimating the cost of risk management explicitly as well as estimating Impact Benefit Cost (savings/profit) of implementing explicit risk management with the agile methods.

3. MATERIALS AND METHODS

A combined list of top ten risks and their corresponding size of loss were extracted from some Agile software development projects conducted in the studies of [6], [14] and [13]. These risks were identified during software development projects. The size of loss is stated in days and for the purpose of this study converted to hours. Size of loss indicates the number of hours required to rework any task(s) that is/are affected by a risk if it actually occurs. The risks are written in their order of priority. Table 1 displays the ten identified risks.

Table 1: Extracted Ten Top Risks

s.no	Ten Top Risks	Size of Loss (in days)	Size of loss (hrs) [days * 8 hours]
1.	Requirement currently unclear	25	200
2.	Lack of experience in creation of user stories	24	192
3.	User and service documents may not be translated into proper language	21	168

4.	The lack of scientifically relevant sample data impacts Partner A's ability to validate the product	20	160
5.	Partner A may require more end user documentation than has been provided	20	160
6.	Backup and restore may require the inclusion of additional 3rd party products	15	120
7.	Product Managers located remotely	12	96
8.	Some team members not assigned full time	12	96
9.	Technology issues such data access	10	80
10.	Partner A employees are not available to validate the new features until too late in the process limiting ability of teams to make additional releases that address any issues that might uncover	5	40
			1300

A list of explicit risk management expenses excluding risk cost expenses alongside the cost of items and services that bring about the expenses are also presented in table 2. The compilation of this list was guided by the explicit risk management constituents presented in the study by [1] in which explicit risk management expense composition consists of risk cost, infrastructural expenses, risk management-related training expenses, risk repository expenses and risk manager's salary expenses. Through market survey in Nigeria by the researchers, the costings were made. Using the purchasing power parity converter, the money was converted to US dollars. This was done since the US dollar is a more acceptable currency worldwide. The values may look under or overestimated because of the conversion. However, it is only used to prove the workability of the model developed. Table 2 displays the risk management expenses excluding risk cost.

Table 2 : Estimated Risk Management Expenses Other than RiskCost

s/ n	Description	Cost (\$)		Cost/yr (\$)
		Cost for 3 yrs	Cost/3 yrs	
1	Purchase of Laptop/Desktop	1200	400	400
2	Purchase of external Backup Hard disk (other form of backup could be used)	120	40	40
3	Purchase of Printer	300	100	100
4	Development of Risk Repository System	850	283	283
5	Training	620		620
6	Risk Manager Remuneration	1600/month		6400
7	Repairs/Services/Maintenance	320		320
8	Toner/ Paper	270		270
Total/yr				8433

4. MODEL FORMULATION

The New Mathematical Risk Management Model (NMRMM) is built to address software development risks associated with Agile software development projects. It consists of nine steps just like the MRMM. The steps include Risk identification, Risk analysis, Total risk amount calculation, Total risk amount calibration, Risk cost evaluation, Requested risk management budget (RRMB) determination, Actual risk management budget and Impact Benefit determination, Risk removal process and Iteration results analysis. Also, it consists of two special CASE tools namely Risk Management Expenses Sheet (RiMEHS) and Agile Risk Information Track Sheet (ARITS). RiMEHS is designed to record all expenses excluding risk cost. These expenses include infrastructural expenses, risk repository development and maintenance expenses, training costs and risk manager remuneration. ARITS (adopted and modified form of ARIS of MRMM) is designed to record identified risk details. It also consist of a risk repository system. The major differences of NMRMM from MRMM are in steps 6 and 7 where requested risk management budget and Actual risk management budget replaces requested risk budget and real risk budget of MRMM respectively.

The abbreviations/notations/acronyms and what they represent in the new model are displayed in table 3.

Table 3: The Abbreviations/Notations/Acronyms and What They Represent in the New Model.

Notation/ Abbreviations	Meaning
AcRMB	the actual risk management cost which is the cost for the risk management process
IF	Infrastructural Budget
IE	Infrastructural Expenses
RMRB	Risk Management Repository Budget
TB	Risk Management-related Training Budget
RMS	Risk Manager Salary
ARMB	Assigned Risk Management Budget
RRMB	Requested Risk Management Budget
AcRMB	Actual Risk Management Budget
OE	OtherExpenses
RRB _T	Requested Risk Budget _T for all risks in all the iterations
RRB _j	Requested risk budget for iteration j
RC _j	Risk Cost of all risks in a particular iteration j
RC _{ij}	Risk Cost of a risk i in a particular iteration j
EC _i	Expected Cost

EC_{ij}	Expected Cost of a risk i in an iteration j
Fee_{ss}	hourly fee to a Semi-skilled manpower effort
Fee_s	hourly fee to a Skilled manpower effort
Fee_{hs}	hourly fee to a High-skilled manpower effort
ARB	Assigned Risk Budget
ARB_{ji}	Assigned budget to any particular risk i in an iteration j
ARB_j	Assigned budget to all risk in an iteration j
ARB_T	Assigned risk budget for all risks in all iterations and is obtained after all iterations.
TRA_j	Total Risk Amount is the sum of biased risk factor
BRF	Biased Risk Factor
BRF_{ji}	Biased Risk Factor for risk i in iteration j
BRF_j	Biased Risk factor in iteration j
SoP_T	Total size of projects
SoP	Size of project
NC	Needed Cost of control is cost for managing
U	UncontrolledLoss
$H_{(k, ss)}$	Total amount of rework hours for all risks taken care of by semi-skilled people
RF_{ij}	Risk factor in iteration j for risk i
H_s	High-skilled software developer
S	Skilled software developer
S_s	Semi-skilled software developer
NOP_{total}	Total number of people involved in the process i.e. hs, s, ss
$NOP_{j,hs}$	number of high-skilled (hs) people in iteration j
NOP_{j,s_2}	number of skilled people in iteration j
$NOP_{j,ss}$	number of semi-skilled people in j
N	Normalized Coefficient
LOC	number of lines of code
$LOC_{j,hs}$	Lines of code written by high-skilled people in iteration j
LOC_{js}	Lines of code written by skilled people in iteration j
LOC_{jss}	Lines of code written by high-skilled people in iteration j
SP_j	Number of story points wriiten in iteration j
SP_{js}	Story points wriiten hy skilled in iteration j
SP_{jss}	Story points wriiten by semi-skilled in iteration j
SP_{total}	Total story points wriiten in iteration j
R	Extent of relevance of the task that will be affected by a particular risk if it occurs
N	Risk removal necessity.
H_{ss}	Total number of hours expended by all semi-skilled developers
H_s	Total number of hours expended by all skilled developers

H _{hs}	Total number of hours expended by all high-skilled developers
SoP	Size of project
NC	Needed Cost of control is cost for managing risk management

4.1 Steps and Equations in the NMRMM Model

Pictorial representation of the steps of the new improved model is shown in figure 1

This includes the following:

1. **Step 1 Risk Identification:** Identifying potential risks. Except for the first iteration, risks listed in each iteration consist of possible risks of previous iteration.
2. **Step 2 Risk Analysis:** This involves studying and understanding the risks identified to know the likelihood of occurrence and impact of the risk. This step is further divided into four stages.

Stage 1, all tasks that are affected by the identified risks are recorded.

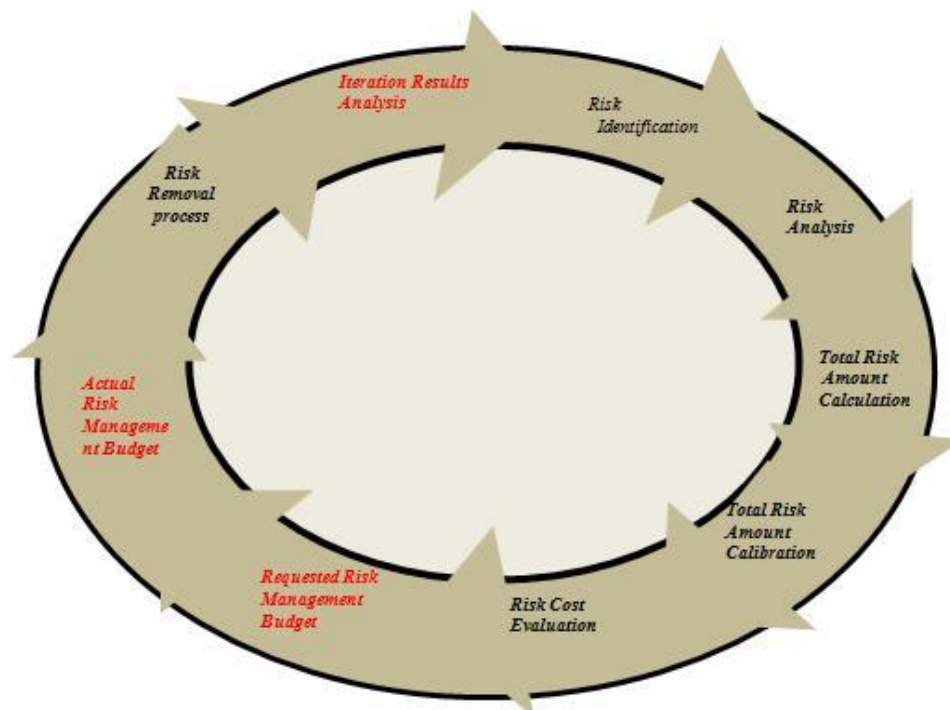


Figure 1: Pictorial representation of the steps of the NMRMM model

b. Stage 2: the level of relevance of each task (task relevancy r) is considered high or low. This is further used to rate or compute the weight of the risk by prioritizing the risks. Knowledge obtained is then used to know whether it is necessary to remove the risks (risk removal necessity n). Thus, risks with higher priority mean that their presence would affect system accuracy more.

c. Stage 3: risk factor amount is made. Risk factors are the uncertain condition influences that will affect the cost, duration and quality of the project negatively (Bannerman, 2008). Each risk

factor has a description and a value. $h(i, k, ss)$ is the amount of rework hours for risk i in iteration j where k is the counter variable for j by a semi-skilled (ss) person.

$$\text{Risk factor } i, j = \frac{\sum_{k=1}^{k=j-1} K \times [h_{(i,k,ss)} \times N_{(k,ss)} + h_{(i,k,s)} \times N_{(k,s)} + h_{(i,k,hs)} \times N_{(k,hs)}]}{\sum_{k=1}^{k=j-1} K \times [H_{(k,ss)} \times N_{(k,ss)} + H_{(k,s)} \times N_{(k,s)} + H_{(k,hs)} \times N_{(k,hs)}]} \quad (1)$$

Just like in the MRMM, The ranking of software developers (development team members) is in form of High-skilled (hs), skilled(s) and semi-skilled(ss) and this in turn judged by the number of hours they can code a particular software feature or user story correctly. The ranking of people involved in the needed number of hours is used to calculate N . N is used here just for the first iteration. The calibration of N is based on historical and environmental data formed from the metrics of other iterations and it is used for the first iteration. Thus, $N = 1$ for semi-skilled (ss), $N = 2$ for skilled (s) and $N = 3$ for High-skilled (hs)

Also, in iteration j , criteria for each identified risk i determination is as follows:

- Experience obtained from similar projects determines how the sums are used. Thus, risk considered is from the second iteration.
- All risks in first iteration to the iteration $(n - 1)$ is considered when risk factors in iteration (n) is determined.
- For each iteration, risk management normalized hours multiplied by the number of that iteration results in weight gain in the risk management normalized hour. This is with respect to the level of closeness of that iteration to the current iteration. The weight of each iteration k , is the variable that displays the iteration

For the rest of the iterations, Calibrated Normalized Coefficient (Normalized amount) is used. It is calibrated based on the historical and the environmental data that formed the following metrics for the other iterations and it is obtained either by using the line of code or story point:

Use of LOC: The number of lines of code (LOC) written by each category of people is the metrics

$$\frac{LOC_{j,hs}}{LOC_{j,total}} \times \frac{NOP_{j,hs}}{NOP_{total}} = n_{j,hs} \quad (2)$$

$$\frac{LOC_{j,s}}{LOC_{j,total}} \times \frac{NOP_{j,s}}{NOP_{total}} = n_{j,s} \quad (3)$$

$$\frac{LOC_{j,ss}}{LOC_{j,total}} \times \frac{NOP_{j,ss}}{NOP_{total}} = n_{j,ss} \quad (4)$$

$$LOC_{j,hs} \approx NOP_{j,hs} \quad (5a)$$

$$LOC_{j,s} \approx NOP_{j,s} \quad (5b)$$

$$LOC_{j,ss} \approx NOP_{j,ss} \quad (5c)$$

It is worthy to note that the proposed NMRMM is flexible in the use of the metrics here.

Use of Story Point: In this paper, story point is a second metrics option for sizing software feature as it is used frequently in agile development. Story points (SP) is used to measure user story and project size and judge skill level. The number of error-free story points of a given user story at a stipulated time is used to judge the level of skill people. On average for the three categories of workers specified, the calibrated N in iteration j is as follows:

$$\frac{SP_{j,hs}}{SP_{j,total}} \times \frac{NOP_{j,hs}}{NOP_{total}} = n_{j,hs} \quad (6)$$

$$\frac{SP_{j,s}}{SP_{j,total}} \times \frac{NOP_{j,s}}{NOP_{total}} = n_{j,s} \quad (7)$$

$$\frac{SP_{j,ss}}{SP_{j,total}} \times \frac{NOP_{j,ss}}{NOP_{total}} = n_{j,ss} \quad (8)$$

$$SP_{j,hs} \approx NOP_{j,hs} \quad (9a)$$

$$SP_{j,s} \approx NOP_{j,s} \quad (9b)$$

$$SP_{j,ss} \approx NOP_{j,ss} \quad (9c)$$

Step 3: Total Risk Amount Determination

$$TRA_j = \sum BRF \quad (10)$$

meanwhile for risk i in iteration j

$$BRF_{j,i} = w(r_j, n_j) \times RF_{ji} \quad (11)$$

- Determination of weight of a risk factor w

$$w = f(r, n)$$

Just like in the MRMM model, As a discrete function between 0 and 2;

$$\left. \begin{aligned} W(r, n) &= 2 \text{ when } r = 1 \text{ and } n = 1, \\ W(r, n) &= \text{when } r = 1 \text{ and } n = 0, \\ W(r, n) &= \text{when } r = 0 \text{ and } n = 0 \text{ or } 1 \end{aligned} \right\} \quad (12)$$

In addition to this, in this paper, consideration has been made where a particular risk has the potentials of affecting more than one task. In that case, relevancy of task is the average value of relevancy of tasks involved. e.g for two tasks t1 and t2 having

$$r = 1 \text{ each, } r = (1 + 1)/2 = 1;$$

$$\text{with t1 and t2 having } r = 1 \text{ and } r = 0 \text{ respectively, } r = (1 + 0)/2 = 0.5$$

$$\text{if } r = 1, \text{ for task t1 and t2,} \quad (13a)$$

Then

$$r = 1 \text{ if } 0.5 \leq r \leq 1 \text{ and } r = 0 \text{ when } 0 \leq r \leq 0.5 \quad (14b)$$

Step 4 Total Risk Amount Calibrations

This is dependent on the particular Agile software development team, level of expertise on the domain knowledge and also on model utilized. Every team uses a tailored version of the particular Agile method utilized. for instance, some agilist utilize colours to imply High Risk, Medium and Mitigated Risk and thus Risk amount calibration.

Step 5 Evaluation of Risk Cost

The total cost of risks in a particular iteration is obtained as follows:

$$RC_j = EC_j \times TRA_j \quad (15)$$

Where

$$EC_j = H_{ss} \times Fee_{ss} + H_s \times Fee_s + H_{hs} \times Fee_{hs} \quad (16)$$

$$\text{RiskCost per Risk} = E_{ci}/E_{cj} \times RC_j \quad (17)$$

$$EC_{ij} = h_{ss} \times fee_{ss} + h_s \times fee_s + h_{hs} \times fee_{hs} \quad (18)$$

Step 6 Determination of Requested Risk Management Budget (RRMB)

The requested risk management budget is made up as follows:

$$RRMB = RRB_T + OE \quad (19)$$

RRMB is obtained after all iterations are completed.

$$RRB_j = \sum RC_j \quad (20)$$

$$RRB_i = RC \text{ per Risk} \quad (21)$$

$$\text{Therefore, } RRB_T = \frac{\sum \text{RiskCost}_t}{\text{Total}} \quad (22)$$

$$OE = IB + RMRB + RTB + RMS \quad (23)$$

$$RRMB = RRB + OE \quad (24)$$

- Estimated Infrastructural Budget (IB) for a particular project will be

$$IB = \frac{\text{SoP}}{\text{SoP}_T \text{ in 3 yrs}} \times \frac{\text{IB}_T / 3 \text{ yrs}}{1} \quad (25)$$

- Estimated Risk Repository Budget (RRB) for a particular project will

$$\text{be } RRB = \frac{\text{SoP}}{\text{SoP}_T / 3 \text{ yrs}} \times \frac{\text{RRB}_T / 3 \text{ yrs}}{1} \quad (26)$$

- Estimated Training Budget (TB) for a particular project will be

$$TB = \frac{\text{SoP}}{\text{SoP}_T / \text{yrs}} \times \frac{\text{TB}_T}{1} \quad (27)$$

- Estimated Risk Manager Salary Budget (RMS) for a particular project will be

$$RMS = \frac{\text{SoP}}{\text{SoP}_T / \text{yr}} \times \frac{\text{RMS}_T / \text{yrs}}{1} \quad (28)$$

Step 7 Determination of Actual Risk Management Budget (ARMB) and Impact Benefit Cost

Risk management is applied based on assigned or allocated budget and this budget is different from the requested risk budget. Thus, Assigned Risk Management Budget will be

$$ARMB = ARB_T + OE \quad (29)$$

The assigned budget to any particular risk *i* in iteration *j* is

$$ARB_{ji} = \frac{BRF_{ji}}{TRA_j} \times \frac{ARB_j}{1} \quad (30)$$

where

$$ARB_j = \frac{EC_j}{EC_T} \times ARB_T \quad (31)$$

But

OE = IB + RMRB + RTB + RMS as in eq 23

ii. Determination of Explicit Risk Management Process
this is the actual risk management budget (AcRMB)

$$AcRMB = OE + NC + \Sigma U \quad (32)$$

$$\text{where } RMPC = ARMC \quad (33)$$

$$NC = \Sigma ARB_T \quad (34)$$

$$U/\text{risk} = RRB/\text{risk} \quad (35)$$

ii. Determination of the Impact Benefit (savings) of Formal Risk Management Practice for the Agile software development process

Impact Benefit of implementing a formal risk management practice in Agile software development (I) will be measured in terms of cost and this will be obtained as follows:

$$I = RRMB - AcRMB \quad (36)$$

Step 8: Risk Removal Process

This includes and follows the order of risk management, risk mitigation and risk control just as found generally in risk management models. However, in this paper, the values of the weight of the risks and how relevant the affected tasks are, guide the plan of actions taken. It includes control measures implemented to prevent or bring the effect of risks to a tolerated minimum on the tasks affected. CASE tools such as the ARITS, RIMEHS and the Risk repository system of the earlier study by the present researchers are used. This will be helpful for maintaining the characteristic swiftness of the agile method as well as serve historical purposes.

Step 9 Iteration Results Analysis: Results of the iteration just concluded are analysed and any unconcluded risk removal process is carried forward to the next iteration.

4.2. Description of Constituents of Explicit Risk Management Process

- i. RiskCost entails what it will take (manpower effort) to rework or correct a risk situation if the risk occurs.
- ii. Other Expenses (OE) are all expenses likely to be incurred during an explicit risk management process excluding risk cost these include infrastructural expenses, training expenses and risk repository system expenses as well as risk manager salary.
OtherExpenses of Risk Management

- i. Infrastructural Expenses: this refers to devices/accessories required to undertake the risk management process. This includes a separate computer system, backup device (secondary storage or cloud), printer, its maintenance and accessories. In this paper, the average life span of computer devices is taken to be 3-4 years and so infrastructural expenses is divided into three years and the result of a year is further shared amongst the major projects of the year using their project size to determine the shared value per project. Thus, the bigger the size of a project, the bigger the cost of risk management apportioned to it.

- ii. Risk Management Repository Budget is budget for developing and maintaining the risk repository system. It is an automated repository where risks records during the agile development are entered into the repository system, computed where necessary and stored into the database.

- iii. Risk Training Budget includes training cost for agile team on the use of the risk management system, this training cost may come up if for instance an old development team member leaves the firm and/or new team member is engaged. Generally it includes all risk management-related trainings for members of the Agile Team and the Risk Manager where the need arises.

- iv. Risk Manager Salary

This is money used for payment for the regular engagement of a Risk Manager who monitors and carries out risk management activities in a dedicated manner.

- Agile Risk Information Track Sheet (ARITS) is the sheet used to record each identified risk.
- Risk Management Expenses Sheet (RiMEHS) is a special Computer Aided Software Engineering (CASE) tool used to record all infrastructural expenses, risk repository development and maintenance expenses, training costs and risk manager remuneration. It is used at anytime in the risk management stages when the need arises. Both ARITS and RIMEHS are CASE tools adopted from previous study of the present researchers.

- A story point is a relative measure used in agile methods to estimate the effort required to develop a software feature or user story and it is a function of project size and complexity. Because it is a relatively estimated feature size by one developer team, it may be different

from another. Thus the size in LOC of the user story used as base measure should be known so as to estimate any user story size in LOC and or project size by multiplying number of SPs in project by LOC of base- user story

- Uncontrolled Loss comprises all monies used for contingency plans. Some identified risk which may be classified as not too necessary to deal with at the beginning may prove otherwise and it may be necessary to rework task(s) affected by that risk afterwards. This will attract another sum of money outside the assigned risk management budget from the software owner.

- Needed Cost of control comprises all the monies used for risk management actions like mitigation of risks before they actually occur and it is thus all monies spent to plan, monitor and mitigate risks. It does not include monies spent on unmanaged risks.

4.3 Difference of Proposed NMRMM and MRMM estimate

Table 4 highlights the difference between the proposed NMRMM and MRMM.

Table 4: Difference between the proposed NMRMM and MRMM.

NMRMM	MRMM
Developed to implement the budget of an entire explicit risk management process	Developed to implement budget of risk cost
Developed to estimate Impact Benefit (in terms of cost) of an entire explicit risk management process	Do not implement Impact Benefit cost estimation as it lacks the necessary parameters to do so
Utilization of a dedicated risk manager	No dedicated risk manager was utilized
Consists of two special CASE tools namely ARITS and RIMEHS	Consists of one special CASE tool namely ARITS
Consists of a special risk repository system (made of an application and a database)	Utilizes a relational database
Utilizes project size metrics of both lines of code and story point	Utilizes project sizing with lines of code metrics Only

5. NMRMM TEST AND VERIFICATION

In other to test the system developed, the top ten risks identified during XP and SCRUM Agile Software Development listed in table 1 is used. Also, inquiry from IT vendors and developers was made on needed infrastructural items and their prices, cost of developing the repository system, possible cost of training, and monthly pay for risk manager tagged Other Expenses were made. Furthermore, it was assumed that 4 major projects with sizes 142 story points (SP), 18 SP, 37 SP and 40 SP were executed in 2016 by a firm. Enterprise Schools Management System with 142 SP project size developed in 3 iterations is used for the test in this paper. The number of hours effective work takes place per day is taken as 8 hours.

First, numerical computations were done and results recorded. Similarly, the values for computations were also entered into the repository systems and results obtained were in line with

those obtained when numerical computation was done. Though results obtained will be discussed here, details of the development of the risk repository system will be made in our further study. Meanwhile, the procedures of managing the risk were followed. Summary of results obtained are shown in Table 4.

6. DISCUSSION OF RESULTS

From the results obtained as shown in table 5 , ordinarily an estimated budget of \$90213.10 will be requested to take care of the 10 risks identified if a well organised risk management process was not carried out but because of putting formal risk management in place, 8 of the identified risks were mitigated with the estimated assigned sum of \$15007 and the remaining 2 identified risks which were left unmanaged eventually manifested and so affected tasks were reworked with the estimated sum of \$10795.10 (UncontrolledLoss). It is worthy to note that while in the existing model-MRMM as shown in the result in table 5, the requested risk management budget consist of only money estimated for riskcost (\$90213.10) , that of the proposed NMRMM consist of Riskcost (\$90213.10) and Otherexpenses (\$4993) which sum up to \$95213.10. In the same vein, the Assigned Budget in MRMM consist of estimate for Riskcost only (\$15007) while that of the proposed NMRMM consist of Riskcost and Otherexpenses which sum up to \$20000. The Actual Risk Management Budget estimate consist of RiskCost (\$15007) alone in MRMM, while that of NMRMM consist of Riskcost (\$15007), OtherExpenses (\$4993) and UncontrolledLoss (\$10795.10) which sum up to \$30695.20 as the result reveals.

UncontrolledLoss is not included in the requested and assigned risk management budget since it is expenses that will result for unmanaged risk cases which seem to affect tasks that are initially considered not too important or whose likelihood of occurrence was judged slim but that actually occurred and needed to be corrected later on. Meanwhile Savings of RiskCost estimate alone is the same in MRMM and NMRMM (i.e \$64411 (71% of the Requested Risk management Budget estimate for RiskCost).

But this is different from the Impact Benefit estimate of Formal Risk Management in the Agile process in terms of Cost (SAVINGS/PROFITs) which is attainable only in NMRMM and has its value as \$64428 (68% of (Total Requested Risk Management Budget estimate)). Formal Risk Management Cost Estimate and Impact benefit of formal risk management is unknown in MRMM as only Riskcost is considered.

Graphical representation of results obtained are shown in figure 2-4.

Table 5: Comparison of Results Obtained in MRMM and NMRMM

Description of Component	Budgetary	MRMM	NMRMM
Software project cost estimate		\$100000	\$100000
Total Requested Risk Management Budget		RiskCost (\$90213.1)	RiskCost (\$90213.1) OtherExpenses (\$4993) \$95213.1
Assigned Risk Management Budget		Risk Cost \$15007	RiskCost (\$15007) OtherExpenses (\$4993) \$20000
Actual Risk Management Total RiskCost		\$25802	\$25802
Actual Risk Management Budget		\$15007	RiskCost (\$15007) OtherExpenses (\$4993) UncontrolledLoss (\$10795) \$30785
Formal Risk Management Process Cost estimate		Not Implemented in MRMM	\$30785
Impact Benefit estimate of RiskCost (SAVINGS/PROFIT)		\$64411 (71% of the Requested Risk management Budget for RiskCost in both MRMM and NMRMM)	
Impact Benefit estimate of Formal risk Management in the Agile process in terms of Cost (SAVINGS/PROFIT)		Not Implemented in MRMM	\$64428 (68% of (Total Requested Risk management Budget estimate))

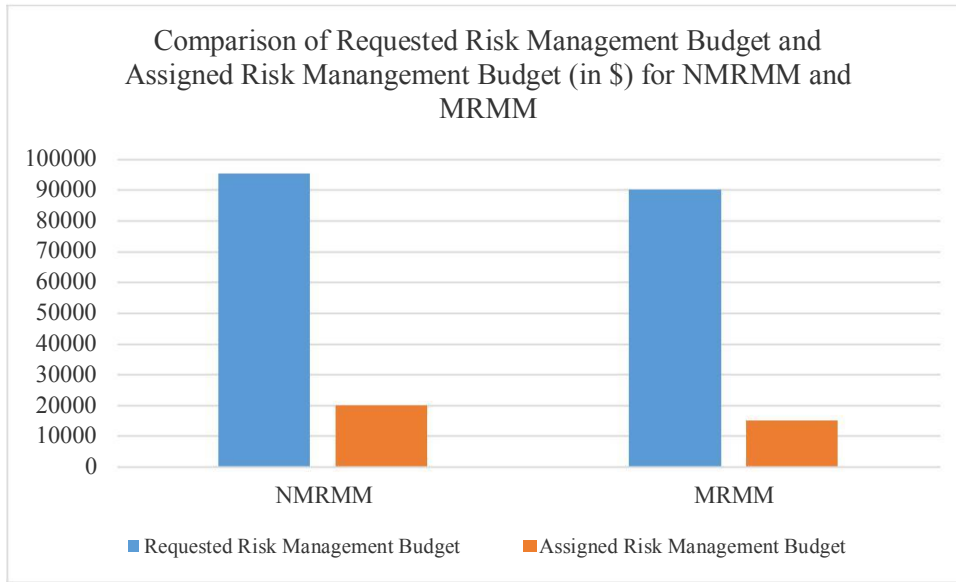


Figure 2: Comparison of Requested Risk Management Budget AND Assigned Risk Management Budget for MRMM and NMRMM

In figure 2, the blue bars represents the Requested Risk Management Budget while the orange bars represents the Assigned Risk Management Budgets for MRMM and NMRMM. The two budgets –Requested and Assigned budgets for NMRMM are higher than those of MRMM. The reason is that in MRMM only Riskcost- a sub-section of risk management is considered while in NMRMM Riskcost, Otherexpenses and UncontrolledLoss which make up an entire risk management process expenses is taken care of.

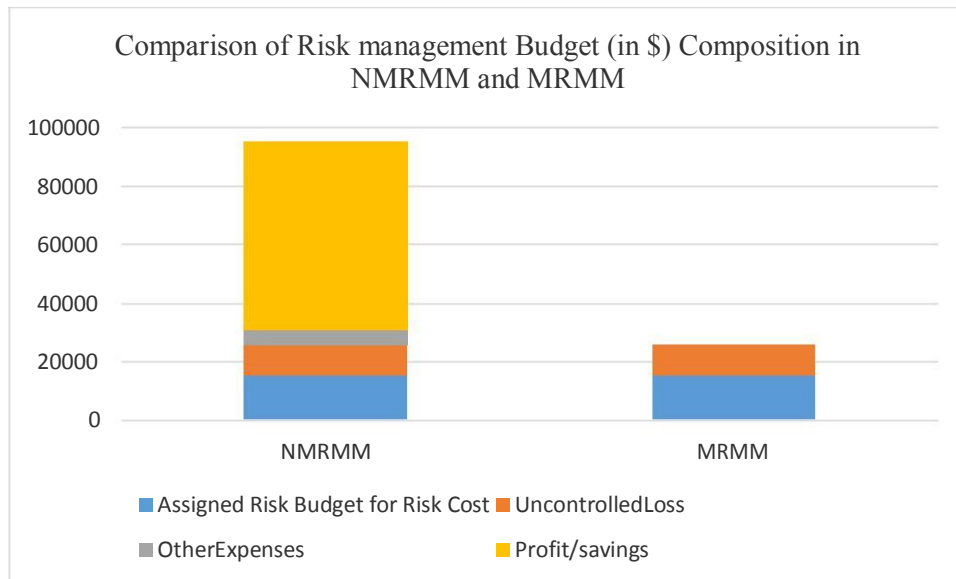


Figure 3: Comparison of risk Management Budget Composition in MRMM and NMRMM

Here in figure 3, the stacked bar chart shows detailed estimates and composition of how risk management budget in MRMM and NMRMM is made up. It can be seen from the bar chart that NMRMM consist of four estimates namely assigned budget for Riskcost, Otherexpenses, UncontrolledLoss and Profit (Impact Benefit of Formal risk management in terms of cost). On the other hand, MRMM consist of Assigned Risk Management for Riskcost and Profit.

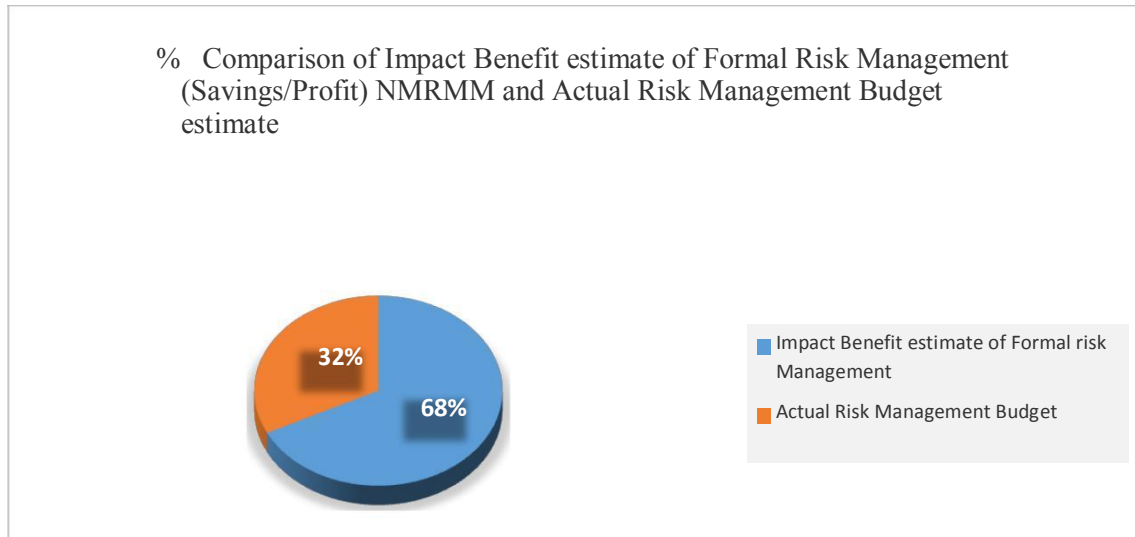


Figure 4: Comparison of Impact Benefit Estimate of Formal Risk Management and Actual Risk Management Budget Estimate in the Agile Development Process

Figure 4 shows that impact benefit of implementing formal risk management is estimated to 68% while 32% was the actual amount spent for managing the risk and the risk management process as a whole.

Consequently, both results obtained in the proposed NMRMM and the existing MRMM risk management model shows that riskcost estimate is achievable and has some cost benefits. However, while the existing MRMM can estimate risks cost alone, the proposed NMRMM can estimate riskcost, risk management process cost and the impact benefit of implementing a formal risk management process. In figure 4 the actual risk management budget estimated as money spent on every risk management activity is 32%. It means only 32% of the requested budget for managing risk was spent for implementing risk management. This in turn means that 68% is unspent and thus savings are made from implementing formal Risk Management. This 68% is the Impact Benefit of Implementing risk management.

This is significant enough to consider implementing risk management explicitly in the agile process. It indicates that the impact benefits derived from implementing formal risk management in the agile process outweighs the implicit or informal way utilized at present in the agile cycle. The percentage difference revealed in the study by the results obtained in the proposed NMRMM (68%) as impact benefit for Implementing Formal Risk Management and existing MRMM (32%) as impact benefit of managing risks is not surprising because MRMM was designed to take care of riskcost alone which does not capture an entire risk management cost estimate and thus does not measure the true value of Impact Benefit of risk management with the agile software

development method. Simply put, the Impact benefit of 71% for RiskCost estimate in MRMM cannot be compared with the Impact Benefit for Formal Risk Management Estimate of 68% in NMRMM as they are measured relative to different quantities.

7. CONCLUSION

Explicit Risk Management implementation in the context of Agile Software Development Methodology was researched and a new Mathematical Risk Management Model with features of Risk Cost estimation, Risk Management Cost estimation and Impact Benefit estimation made. This is necessary as Agile methods though eliminate some risk issues using their inherent risk management approach, do not follow the necessary guidelines of a true risk management procedures and as a result is capable of by-passing some risks conditions unaddressed. This occurs especially in large projects where numbers of iterations are typically large. Generally, the implementation of explicit risk management in the agile software projects will help identify most of the risks likely to threaten the smooth actualization of the project goal and also, help to monitor and mitigate most of the risks identified. Specifically, from the findings obtained, explicit risk Management process estimation is doable, less costly to implement and its Impact Benefit (in terms of cost) is reasonable enough to encourage stakeholders. Consequently, risk management budget and its impact benefit cost estimation being novel contributions in this area, will help stakeholders to have knowledge of implementing explicit risk management and its impact benefits. Hence, those initially sceptical about implementing it will know that it is worth doing. With the involvement of a dedicated risk manager, all identified risks are recorded and managed and the agile development process is not interrupted in any way. The CASE tools used will facilitate computations, no matter the number of risks and iterations in the project, further maintaining the agility of the agile methods used as well as making it suitable for all sizes of projects.

References

- [1] Thom-Manuel, O. M, Ugwu, C. & Onyejegbu (2017) “An Extended Agile Software Development Project Budget Model”. *International Journal of Computer Science and Software Engineering (IJCSSE)*, 6(12):306-314
- [2] Nyfjord J, Kajko-Mattsson M (2008) “Integrating Risk Management with Software Development: State of Practice.” In : *Proceedings of the International Conference on Engineers and Computer Scientists, Vol. I*, pp 878–884
- [3] Islam, S. (2010) “Software Development Risk Management Model- a goal-driven approach”, Ph.D Thesis, Institute für Informatik, Technische Universität München, Germany, 2011.
- [4] Wiegers, E. K (2007) “Practical Project Initiation: A handbook with Tools” , Know your enemy. 77-88
- [5] Kontio, J. (2001) “Software Engineering Risk Management: A Method, Improvement Framework, and Empirical Evaluation” Ph.D Thesis. Retrieved from <http://lib.tkk.fi/Diss/2001/isbn951225655X/isbn951225655X.pdf>
- [6] Cohn, M. (2010) “Succeeding with agile: software development using Scrum”. Pearson Education.
- [7] Highsmith, J. & Cockburn, A. (2001) “Agile software development: The people factor”. *IEEE Computer* 34(131-133).
- [8] Alharbi, E. T., Rizwan, .M. & Qureshi, J. (2014) “Implementation of Risk Management with SCRUM to Achieve CMMI Requirements”, *Computer Network and Information Security*,11: 20-25.
- [9] Derfer, B. (2016) “Introducing the Agile Risk Management Framework, Agile Six Applications, Inc”. Retrieved from https://www.agilegovleaders.org/wpcontent/uploads/2016/03/Agile_Risk_Management_Framework.pdf
- [10] Ten Six Consulting. (March 28, 2017). 7 Impacts of Poor Risk Management (And What You Can Do About Them) [Blog post]. Retrieved from <https://tensix.com/2017/03/7-impacts-of-poor-risk-management-and-what-you-can-do-about-them/>
- [11] Khatavakhotan, A. S. (2011) “MRMM: A Mathematical Risk Management Model for Iterative IT Projects based on the Smart Database.” *International Journal of Information and Electronics Engineering*, 1(3), 229-233
- [12] Tomanek, .M. & Juricek, J. (2015). “Project Risk Management Model based on PRINCE2 and Scrum Frameworks.” *International Journal of Software Engineering & Applications (IJSEA)*, 6(1). DOI: 10.5121/ijsea.2015.6107 81
- [13] Mathkour, H. Assassa, G. & Baihan, A. (2008) “A Risk Management Tool for Extreme Programming.” *International Journal of Computer Science and Network Security* 8 (8)
- [14] Nyfjord , J. (2008). *Towards Integrating Agile Development And Risk Management*. Stockholm University , Department of Computer and Systems Sciences . Sweden: Universite tsservice US-AB, Kista. Retrieved from <https://www.diva-portal.org/smash/get/diva2 :199663/FULLTEXT01.pdf>
- [15] Veehil, S. T. (May 3, 2013) *Risk Management in Agile*. Retrieved from SCRUM ALLIANCE, Inc: <https://www.scrumalliance.org/community/articles /2013/2013-may/risk-management-in-agile>
- [16] Bannerman, P. (2008). “Risk and risk management in software projects: A reassessment.” *The Journal of Systems and Software* , 81, 2118-2133